

UNITED STATES
PATENT APPLICATION

for

**COPYING A PORTION OF A DATABASE
STRUCTURE ASSOCIATED WITH A QUERY**

NCR Docket No. 9928

submitted by

**Jerry L. Klindt
Paul L. Sinclair**

on behalf of

**NCR Corporation
Dayton, Ohio**

Express Mail No.: EL609946512US

Date: June 27, 2001

Prepared by

Howard L. Speight
Reg. 37,733

Correspond with

John D. Cowart
Reg. 38,415
Teradata Law IP, WHQ-4W
NCR Corporation
1700 S. Patterson Blvd.
Dayton, Oh 45479-0001
(858) 485-4903 [Voice]
(858) 485-2581 [Fax]

COPYING A PORTION OF A DATABASE STRUCTURE ASSOCIATED WITH A QUERY

Background

[0001] Database system providers commonly offer technical support to their customers. As part of this support, the providers assist customers in troubleshooting problems that arise in customer 5 databases, for example, when a SQL query fails. SQL is the dominant language for managing information in relational databases.

[0002] Troubleshooting a failed query can be difficult, since it is often necessary to attempt to reproduce the problem, possibly in the system provider's laboratory. This may require the system provider to retrieve from the customer the definitions of any objects associated with the query, 10 including definitions of objects that are in turn referenced by these definitions. It also may require capturing configuration information regarding the computer system and a statistical view of data in the database.

[0003] To obtain the definitions, the technical support provider typically identifies each database object associated with the query, for example as a table or a view. The provider then executes an appropriate SQL command for the object, which returns the object definition. For example, in the Teradata Active Data Warehousing System, available from NCR Corporation, a SHOW TABLE command returns the definition of a table in the form of the data definition language (DDL) necessary to create the table. Similarly, the SHOW VIEW, SHOW MACRO, SHOW TRIGGER, and SHOW JOIN INDEX commands return the definitions of those database objects. The database object 20 definitions are stored in a data dictionary (DD). The provider then identifies each object referenced within the retrieved definition, obtains definitions for those objects, and repeats the process. This continues until the provider believes that all of the necessary object definitions have been obtained. For some database objects, such as triggers and join indexes, the association of the database object with the query may only become apparent upon examination of the data dictionary.

25 [0004] The definitions may then be transferred to a computer in the system provider's laboratory. Possible difficulties arise when insufficient object definitions are obtained, either by the provider or by the customer, either of whom may have insufficiently stated or understood the problem. The provider may discover that definitions were not obtained for objects referenced in the definition of another object, in which case additional information must be obtained from the customer's database system. In 30 a situation such as this, it may take multiple contacts with the customer over a period of several days to

obtain the necessary information. Such multiple contacts may produce customer dissatisfaction with the service provided. Furthermore, the customer may alter one or more object definitions between the time the customer identifies the problem and the time the definitions are gathered. If the customer's alterations eliminate the problem, the service provider may spend a great deal of time trying to solve a problem that no longer exists using the gathered definitions.

Summary

[0005] In general, in one aspect, the invention features a method for copying a portion of a database structure, the database including one or more database objects. The method includes recursively retrieving object definitions for one or more database objects associated with a query to produce an ordered set of object definitions. The method further includes building a copy of the database structure using the ordered set of object definitions.

[0006] Implementations of the invention may include one or more of the following. Recursively retrieving object definitions may include recursively identifying objects associated with the query, categorizing each identified object into a category, and retrieving an object definition for each identified object using a tool corresponding to the category with which the identified object is connected. The categories may include tables, views, join indexes, triggers and macros. The tool may be a SHOW VIEW statement if the identified object is categorized as a view, a SHOW TABLE statement if the identified object is categorized as a table, a SHOW JOIN INDEX statement if the identified object is categorized as a join index, a SHOW TRIGGER statement if the identified object is categorized as a trigger, or a SHOW MACRO statement if the identified object is categorized as a macro.

[0007] Recursively retrieving object definitions may include retrieving unretrieved object definitions for a set of objects known to be associated with the query, adding to the set of objects known to be associated with the query those objects contained in the retrieved object definitions that are not already in the set of objects known to be associated with the query, and repeating these two items until no new objects are added to the set of objects known to be associated with the query.

[0008] The method may include sending the ordered set of object definitions from a first computer to a second computer. The order of the ordered set of object definitions may also be changed. Changing the order may include reordering the object definitions so that each object definition is ordered before

the definition of any object that references it. The object definitions may be ordered so that each object definition is ordered before the definition of any object that references it.

[0009] Recursively retrieving object definitions for one or more database objects may include looking for references to the one or more database objects in a data dictionary.

5 [0010] In general, in another aspect, the invention features a computer-readable medium containing computer-executable code for instructing a computer to recursively retrieve object definitions for one or more database objects associated with a query to produce an ordered set of object definitions. The computer-executable code includes computer-executable code for instructing a computer to build a copy of the database structure using the ordered set of object definitions.

10 [0011] Implementations of the invention may include one or more of the following. The computer-executable code, when recursively retrieving object definitions, may instruct the computer to recursively identify objects associated with the query and categorize each identified object into a category. The computer-executable code may then retrieve an object definition for each identified object using a tool corresponding to the category with which the identified object is connected. The categories may include tables, views, join indexes, triggers and macros. The tool may be a SHOW VIEW statement if the identified object is categorized as a view, a SHOW TABLE statement if the identified object is categorized as a table, a SHOW JOIN INDEX statement if the identified object is categorized as a join index, a SHOW TRIGGER statement if the identified object is categorized as a trigger, or a SHOW MACRO statement if the identified object is categorized as a macro.

15 [0012] The computer-executable code may instruct the computer, when recursively retrieving object definitions, to retrieve unretrieved object definitions for a set of objects known to be associated with the query, and may add to the set of objects known to be associated with the query those objects contained in the retrieved object definitions that are not already in the set of objects known to be associated with the query. The code may repeat these instructions until no new objects are added to the set of objects known to be associated with the query.

20 [0013] The computer-executable code may instruct the computer to send the object definitions from a first computer to a second computer. The code may instruct the computer to change the order of the ordered set of object definitions. The computer may be instructed to reorder the object definitions so that each object definition is ordered before the definition of any object that references it. The

computer may be instructed to store the definitions so that each object definition is ordered before the definition of any object that references it.

[0014] The computer-executable code may instruct the computer, when recursively retrieving object definitions for one or more database objects, to look for references to the one or more database objects in a data dictionary.

[0015] In general, in another aspect, the invention features a package of data useful in building a copy of a database structure. The package is generated in accordance with the act of recursively retrieving object definitions for one or more database objects associated with a query to produce an ordered set of object definitions.

[0016] Implementations of the invention may include one or more of the following. The object definitions may be recursively retrieved in accordance with the acts of recursively identifying objects associated with the query, categorizing each identified object into a category, retrieving an object definition for each identified object using a tool corresponding to the category with which the identified object is connected. The categories may include tables, views, join indexes, triggers and macros. The tool may be a SHOW VIEW statement if the identified object is categorized as a view, a SHOW TABLE statement if the identified object is categorized as a table, a SHOW JOIN INDEX statement if the identified object is categorized as a join index, a SHOW TRIGGER statement if the identified object is categorized as a trigger, or a SHOW MACRO statement if the identified object is categorized as a macro.

[0017] The object definitions may be recursively retrieved in accordance with the acts of retrieving unretrieved object definitions for a set of objects known to be associated with the query, adding to the set of objects known to be associated with the query those objects contained in the retrieved object definitions that are not already in the set of objects known to be associated with the query, and repeating until no new objects are added to the set of objects known to be associated with the query.

[0018] The implementation may include the act of sending the ordered set of object definitions from a first computer to a second computer, as well as the act of instructing the computer to change the order of the ordered set of object definitions. The act of changing the order of the ordered set of object definitions may include reordering the object definitions so that each object is ordered before the definition of any object that references it.

- [0019] The act of storing object definitions may include storing the object definitions so that each object definition is ordered before the definition of any object that references it.
- [0020] The act of recursively retrieving object definitions for one or more database objects may include looking for references to the one or more database objects in a data dictionary.
- 5 [0021] Other features and advantages will become apparent from the description and claims that follow.

Brief Description of the Drawings

- [0022] Fig. 1 is a block diagram of a node of a database system.
- 10 [0023] Fig. 2 is a block diagram of a parsing engine.
- [0024] Fig. 3 is a flow chart of a parser.
- [0025] Fig. 4 is a block diagram of a method for reproducing a database environment.
- [0026] Fig. 5 is a block diagram of an example of the structure of a query and the database elements it references.
- [0027] Fig. 6 is a flow chart of a method for displaying underlying database objects.
- [0028] Fig. 7 is a flow chart of a method for finding and obtaining definitions for database objects associated with a query.
- [0029] Fig. 8 is a flow chart of a method for obtaining object definitions.
- [0030] Fig. 9 is a flow chart of a method for retrieving object definitions.

20 Detailed Description

- [0031] The database object retrieval technique disclosed herein has particular application to large databases which might contain many millions or billions of records managed by a database system ("DBS") 100, such as a Teradata Active Data Warehousing System available from NCR Corporation.
- 25 Fig. 1 shows a sample architecture for one node 105₁ of the DBS 100. The DBS node 105₁ includes one or more processing modules 110_{1...N}, connected by a network 115, that manage the storage and

retrieval of data in data-storage facilities $120_{1\dots N}$. Each of the processing modules $110_{1\dots N}$ may be one or more physical processors or each may be a virtual processor, with one or more virtual processors running on one or more physical processors.

[0032] For the case in which one or more virtual processors are running on a single physical processor, 5 the single physical processor swaps between the set of N virtual processors.

[0033] Each of the processing modules $110_{1\dots N}$ manages a portion of a database that is stored in a corresponding one of the data-storage facilities $120_{1\dots N}$. Each of the data-storage facilities $120_{1\dots N}$ includes one or more disk drives. The DBS may include multiple nodes $105_{2\dots N}$ in addition to the illustrated node 105_1 , connected by extending the network 115.

10 [0034] The system stores data in one or more tables in the data-storage facilities $120_{1\dots N}$. The rows $125_{1\dots Z}$ of the tables are stored across multiple data-storage facilities $120_{1\dots N}$ to ensure that the system workload is distributed evenly across the processing modules $110_{1\dots N}$. A parsing engine 130 organizes the storage of data and the distribution of table rows $125_{1\dots Z}$ among the processing modules $110_{1\dots N}$. The parsing engine 130 also coordinates the retrieval of data from the data-storage facilities $120_{1\dots N}$ in response to queries received from a user at a mainframe 135 or a client computer 140. The DBS 100 usually receives queries in a standard format, such as SQL.

15 [0035] In one example system, the parsing engine 130 is made up of three components: a session control 200, a parser 205, and a dispatcher 210, as shown in Fig. 2. The session control 200 provides the logon and logoff function. It accepts a request for authorization to access the database, verifies it, 20 and then either allows or disallows the access.

[0036] Once the session control 200 allows a session to begin, a user may submit a SQL request, which is routed to the parser 205. As illustrated in Fig. 3, the SQL request is checked for proper SQL syntax (block 300), evaluated semantically (block 305), and checked in the data dictionary to ensure that all of the objects specified in the SQL request actually exist and that the user has the authority to 25 perform the request (block 310). Finally, the parser 205 runs an optimizer (block 315), which develops the least expensive plan to perform the request.

[0037] As depicted in Fig. 4, the process of reproducing a database environment (block 405) may include several steps: obtaining object definitions of data in a database (block 410); capturing a statistical view of data in the database (block 415); and capturing configuration information concerning

the database (block 420). The system described herein automates the step of obtaining object definitions (block 410). The other processes (blocks 415 and 420) are outside the scope of this disclosure. The system provides in one automated procedure a complete list of object definitions. By performing this function, the system can help reduce the time between identification of a problem by a 5 customer and obtaining the object definitions necessary to troubleshoot the problem.

[0038] The automated process of obtaining database object definitions typically begins with a SQL command 505, e.g., SHOW SELECT A, B FROM X, Y, as shown in Fig. 5. The command 505 may contain the failed query, in this case "SELECT A, B FROM X, Y." This type of query is often called a Data Manipulation Language (DML) statement, and command 505 may be called a "SHOW DML 10 statement." Although command 505 is available to all users of the system, for security reasons the user issuing the SHOW DML statement must have permission to access the database objects associated with the query, which may be any object in the database. An object is "associated" with a query if the object appears in the query itself or appears in the definition of an associated object or is somehow associated with the query through the data dictionary. Fig. 5 displays the database objects that are associated with this example query. Table X object 520 is associated with the query because it explicitly appears in the query. Trigger 525 is associated with the query because it appears in the data dictionary as code to be implemented when defined changes occur to data contained in Table X 520, as signified by the line connecting the two boxes. View Y 530 is associated with the query because it is explicitly included in the query 505. The association of a join index 535 is revealed upon examination of the data dictionary. A table 540 and a view 545 are associated with the query because they are included in the definition of View Y 530. Similarly, a table 550 and a view 555 are associated with the query because they are included in the definition of the view 545, and table 560 is associated with the query because it is included in the definition of the view 555.

[0039] In existing technical support environments, the technical support staff will execute successive 25 SHOW statements and queries into the data dictionary to find the definitions of objects associated with the query. The system disclosed herein automates this process by recursively retrieving object definitions for objects known to be associated with a query, and adding to a set of objects known to be associated with the query any new objects identified in the retrieved definitions.

0038
0039
0040
0041
0042
0043
0044
0045
0046
0047
0048
0049
0050
0051
0052
0053
0054
0055
0056
0057
0058
0059
0060
0061
0062
0063
0064
0065
0066
0067
0068
0069
0070
0071
0072
0073
0074
0075
0076
0077
0078
0079
0080
0081
0082
0083
0084
0085
0086
0087
0088
0089
0090
0091
0092
0093
0094
0095
0096
0097
0098
0099
0100
0101
0102
0103
0104
0105
0106
0107
0108
0109
0110
0111
0112
0113
0114
0115
0116
0117
0118
0119
0120
0121
0122
0123
0124
0125
0126
0127
0128
0129
0130
0131
0132
0133
0134
0135
0136
0137
0138
0139
0140
0141
0142
0143
0144
0145
0146
0147
0148
0149
0150
0151
0152
0153
0154
0155
0156
0157
0158
0159
0160
0161
0162
0163
0164
0165
0166
0167
0168
0169
0170
0171
0172
0173
0174
0175
0176
0177
0178
0179
0180
0181
0182
0183
0184
0185
0186
0187
0188
0189
0190
0191
0192
0193
0194
0195
0196
0197
0198
0199
0200
0201
0202
0203
0204
0205
0206
0207
0208
0209
0210
0211
0212
0213
0214
0215
0216
0217
0218
0219
0220
0221
0222
0223
0224
0225
0226
0227
0228
0229
0230
0231
0232
0233
0234
0235
0236
0237
0238
0239
0240
0241
0242
0243
0244
0245
0246
0247
0248
0249
0250
0251
0252
0253
0254
0255
0256
0257
0258
0259
0260
0261
0262
0263
0264
0265
0266
0267
0268
0269
0270
0271
0272
0273
0274
0275
0276
0277
0278
0279
0280
0281
0282
0283
0284
0285
0286
0287
0288
0289
0290
0291
0292
0293
0294
0295
0296
0297
0298
0299
0300
0301
0302
0303
0304
0305
0306
0307
0308
0309
0310
0311
0312
0313
0314
0315
0316
0317
0318
0319
0320
0321
0322
0323
0324
0325
0326
0327
0328
0329
0330
0331
0332
0333
0334
0335
0336
0337
0338
0339
0340
0341
0342
0343
0344
0345
0346
0347
0348
0349
0350
0351
0352
0353
0354
0355
0356
0357
0358
0359
0360
0361
0362
0363
0364
0365
0366
0367
0368
0369
0370
0371
0372
0373
0374
0375
0376
0377
0378
0379
0380
0381
0382
0383
0384
0385
0386
0387
0388
0389
0390
0391
0392
0393
0394
0395
0396
0397
0398
0399
0400
0401
0402
0403
0404
0405
0406
0407
0408
0409
0410
0411
0412
0413
0414
0415
0416
0417
0418
0419
0420
0421
0422
0423
0424
0425
0426
0427
0428
0429
0430
0431
0432
0433
0434
0435
0436
0437
0438
0439
0440
0441
0442
0443
0444
0445
0446
0447
0448
0449
0450
0451
0452
0453
0454
0455
0456
0457
0458
0459
0460
0461
0462
0463
0464
0465
0466
0467
0468
0469
0470
0471
0472
0473
0474
0475
0476
0477
0478
0479
0480
0481
0482
0483
0484
0485
0486
0487
0488
0489
0490
0491
0492
0493
0494
0495
0496
0497
0498
0499
0500
0501
0502
0503
0504
0505
0506
0507
0508
0509
05010
05011
05012
05013
05014
05015
05016
05017
05018
05019
05020
05021
05022
05023
05024
05025
05026
05027
05028
05029
05030
05031
05032
05033
05034
05035
05036
05037
05038
05039
05040
05041
05042
05043
05044
05045
05046
05047
05048
05049
05050
05051
05052
05053
05054
05055
05056
05057
05058
05059
05060
05061
05062
05063
05064
05065
05066
05067
05068
05069
05070
05071
05072
05073
05074
05075
05076
05077
05078
05079
05080
05081
05082
05083
05084
05085
05086
05087
05088
05089
05090
05091
05092
05093
05094
05095
05096
05097
05098
05099
050100
050101
050102
050103
050104
050105
050106
050107
050108
050109
050110
050111
050112
050113
050114
050115
050116
050117
050118
050119
050120
050121
050122
050123
050124
050125
050126
050127
050128
050129
050130
050131
050132
050133
050134
050135
050136
050137
050138
050139
050140
050141
050142
050143
050144
050145
050146
050147
050148
050149
050150
050151
050152
050153
050154
050155
050156
050157
050158
050159
050160
050161
050162
050163
050164
050165
050166
050167
050168
050169
050170
050171
050172
050173
050174
050175
050176
050177
050178
050179
050180
050181
050182
050183
050184
050185
050186
050187
050188
050189
050190
050191
050192
050193
050194
050195
050196
050197
050198
050199
050200
050201
050202
050203
050204
050205
050206
050207
050208
050209
050210
050211
050212
050213
050214
050215
050216
050217
050218
050219
050220
050221
050222
050223
050224
050225
050226
050227
050228
050229
050230
050231
050232
050233
050234
050235
050236
050237
050238
050239
050240
050241
050242
050243
050244
050245
050246
050247
050248
050249
050250
050251
050252
050253
050254
050255
050256
050257
050258
050259
050260
050261
050262
050263
050264
050265
050266
050267
050268
050269
050270
050271
050272
050273
050274
050275
050276
050277
050278
050279
050280
050281
050282
050283
050284
050285
050286
050287
050288
050289
050290
050291
050292
050293
050294
050295
050296
050297
050298
050299
050300
050301
050302
050303
050304
050305
050306
050307
050308
050309
050310
050311
050312
050313
050314
050315
050316
050317
050318
050319
050320
050321
050322
050323
050324
050325
050326
050327
050328
050329
050330
050331
050332
050333
050334
050335
050336
050337
050338
050339
050340
050341
050342
050343
050344
050345
050346
050347
050348
050349
050350
050351
050352
050353
050354
050355
050356
050357
050358
050359
050360
050361
050362
050363
050364
050365
050366
050367
050368
050369
050370
050371
050372
050373
050374
050375
050376
050377
050378
050379
050380
050381
050382
050383
050384
050385
050386
050387
050388
050389
050390
050391
050392
050393
050394
050395
050396
050397
050398
050399
050400
050401
050402
050403
050404
050405
050406
050407
050408
050409
050410
050411
050412
050413
050414
050415
050416
050417
050418
050419
050420
050421
050422
050423
050424
050425
050426
050427
050428
050429
050430
050431
050432
050433
050434
050435
050436
050437
050438
050439
050440
050441
050442
050443
050444
050445
050446
050447
050448
050449
050450
050451
050452
050453
050454
050455
050456
050457
050458
050459
050460
050461
050462
050463
050464
050465
050466
050467
050468
050469
050470
050471
050472
050473
050474
050475
050476
050477
050478
050479
050480
050481
050482
050483
050484
050485
050486
050487
050488
050489
050490
050491
050492
050493
050494
050495
050496
050497
050498
050499
050500
050501
050502
050503
050504
050505
050506
050507
050508
050509
050510
050511
050512
050513
050514
050515
050516
050517
050518
050519
050520
050521
050522
050523
050524
050525
050526
050527
050528
050529
050530
050531
050532
050533
050534
050535
050536
050537
050538
050539
050540
050541
050542
050543
050544
050545
050546
050547
050548
050549
050550
050551
050552
050553
050554
050555
050556
050557
050558
050559
050560
050561
050562
050563
050564
050565
050566
050567
050568
050569
050570
050571
050572
050573
050574
050575
050576
050577
050578
050579
050580
050581
050582
050583
050584
050585
050586
050587
050588
050589
050590
050591
050592
050593
050594
050595
050596
050597
050598
050599
050600
050601
050602
050603
050604
050605
050606
050607
050608
050609
050610
050611
050612
050613
050614
050615
050616
050617
050618
050619
050620
050621
050622
050623
050624
050625
050626
050627
050628
050629
050630
050631
050632
050633
050634
050635
050636
050637
050638
050639
050640
050641
050642
050643
050644
050645
050646
050647
050648
050649
050650
050651
050652
050653
050654
050655
050656
050657
050658
050659
050660
050661
050662
050663
050664
050665
050666
050667
050668
050669
050670
050671
050672
050673
050674
050675
050676
050677
050678
050679
050680
050681
050682
050683
050684
050685
050686
050687
050688
050689
050690
050691
050692
050693
050694
050695
050696
050697
050698
050699
050700
050701
050702
050703
050704
050705
050706
050707
050708
050709
050710
050711
050712
050713
050714
050715
050716
050717
050718
050719
050720
050721
050722
050723
050724
050725
050726
050727
050728
050729
050730
050731
050732
050733
050734
050735
050736
050737
050738
050739
050740
050741
050742
050743
050744
050745
050746
050747
050748
050749
050750
050751
050752
050753
050754
050755
050756
050757
050758
050759
050760
050761
050762
050763
050764
050765
050766
050767
050768
050769
050770
050771
050772
050773
050774
050775
050776
050777
050778
050779
050780
050781
050782
050783
050784
050785
050786
050787
050788
050789
050790
050791
050792
050793
050794
050795
050796
050797
050798
050799
050800
050801
050802
050803
050804
050805
050806
050807
050808
050809
050810
050811
050812
050813
050814
050815
050816
050817
050818
050819
050820
050821
050822
050823
050824
050825
050826
050827
050828
050829
050830
050831
050832
050833
050834
050835
050836
050837
050838
050839
050840
050841
050842
050843
050844
050845
050846
050847
050848
050849
050850
050851
050852
050853
050854
050855
050856
050857
050858
050859
050860
050861
050862
050863
050864
050865
050866
050867
050868
050869
050870
050871
050872
050873
050874
050875
050876
050877
050878
050879
050880
050881
050882
050883
050884
050885
050886
050887
050888
050889
050890
050891
050892
050893
050894
050895
050896
050897
050898
050899
050900
050901
050902
050903
050904
050905
050906
050907
050908
050909
050910
050911
050912
050913
050914
050915
050916
050917
050918
050919
050920
050921
050922
050923
050924
050925
050926
050927
050928
050929
050930
050931
050932
050933
050934
050935
050936
050937
050938
050939
050940
050941
050942
050943
050944
050945
050946
050947
050948
050949
050950
050951
050952
050953
050954
050955
050956
050957
050958
050959
050960
050961
050962
050963
050964
050965
050966
050967
050968
050969
050970
050971
050972
050973
050974
050975
050976
050977
050978
050979
050980
050981
050982
050983
050984
050985
050986
050987
050988
050989
050990
050991
050992
050993
050994
050995
050996
050997
050998
050999
051000
05100

[0040] The automated system recursively retrieves object definitions for database objects associated with the SQL query, as shown in Fig. 6, and stores the object definitions in an ordered set, for example in a file (block 605).

[0041] Frequently, the order of the object definitions within the file will not be convenient, in the sense that the file could not be used to recreate the database in a single submission to the database creation process. In one example system, the object definitions are reordered within the file (block 610) so that, if this file were submitted to the database creation process, the definition of each database object would output before the definitions of any objects that reference it. This is an optional step, because the effect of reordering the object definitions within the file could be accomplished by repeatedly 10 submitting the file to the database creation process (block 620) until no error messages are returned. Each successive submission would create that portion of the database that it was possible to create given the database structure already in place. The submissions would continue until the structure is complete. Alternatively, the object definitions could be manually reordered within the file. In still another alternative, an automated process could be applied to reorder the object definitions within the file.

[0042] In most cases, the file containing the ordered (or reordered, if block 610 is applied) set of object definitions is sent to another computer (block 615) using a computer utility such as email or FTP. This may happen automatically or as the result of human intervention. In one example, the definitions may be obtained on a customer's computer and then transmitted to a system provider's computer. Alternatively, the set of object definitions may be displayed on a screen or saved to a file. In some cases, the file is not sent to another computer and the database is rebuilt on the same computer.

[0043] Finally, the object definitions are used to create a copy of the database structure (block 620) by submitting the file to a database creation process.

[0044] The process of recursively retrieving object definitions for database objects associated with a query, illustrated in more detail in Fig. 7, includes identifying database objects associated with the query (block 705). If any new objects were identified (block 710), object definitions for the new objects are retrieved and added to the set of object definitions associated with the query (block 715). A number of methods for recording which database objects are new are possible. For example, the system may maintain a set of objects known to be associated with the query. The set may be in a file 25 where the database objects are stored as they are discovered. When the system encounters a database 30

object it checks the file. If the database object is not in the file, the system will add the database object to the file of database objects. Furthermore, the ordered set of database object definitions may be a file with the object definitions stored sequentially as they are discovered, in which case the set of objects associated with the query may be the database objects listed in the file of object definitions. Other ways of recording the database object definitions will be apparent without further discussions.

[0045] For the first pass through the iterative process illustrated in Fig. 7, the system will find the database objects explicitly identified in the query or that are tied to those database objects through the data dictionary. For example, in the query 505 illustrated in Fig. 5, the process will find Table X 520 and View Y 530 in the first pass. Since these are new objects (block 710), the system will retrieve their object definitions and start the ordered set of object definitions associated with the query. The system would examine the data dictionary and find that a join index 535 has been created that might be useful in executing the query because it covers all or a portion of Table X 520 or View Y 530. Consequently, the join index would be identified as a database object associated with the query and its definition would be added to the ordered set. Similarly, upon examination of the data dictionary, the system would identify trigger 525 as a database object associated with the query because of its tie to Table X 520. Consequently, the trigger's definition would be added to the ordered set. In subsequent passes, the system searches for new database objects identified in the object definitions in the ordered set (block 705). If any are found (block 710), their object definitions are retrieved and added to the ordered set (block 715). This process repeats until no new objects are found..

[0046] For example, continuing with the structure illustrated in Fig. 5, the system would examine the object definitions present in the ordered set and would find table 540 and view 545 in the definition of View Y 530. Consequently, their definitions would be added to the ordered set. In the next pass, the system would find table 550 and view 555 in the definition of view 545 and their definitions would be added to the ordered set. In the next pass, the system would find table 560 in the definition of view 555 and its definition would be added to the ordered set. In the next pass, no new objects would be found and the process of recursively retrieving database object definitions (block 605 in Fig. 6) would be complete.

[0047] An example of the process of retrieving unretrieved object definitions and adding them to the ordered set (block 715), illustrated in more detail in Fig. 8, begins by categorizing the object (block 805). In one example system, a database object can be a table, a view, a trigger, a macro or a join

index. The system then retrieves the object definition for the database object using a tool corresponding to the object's category (block 810). The system then adds the object definition to the ordered set (block 815). The system repeats this process until no more objects are left (block 820).

[0048] Objects are categorized (block 805) so that the correct tool may be used to retrieve the object's definition (block 810). An example set of possible tools for various object categories (blocks 905-925), illustrated in Fig. 9, includes SHOW TABLE (block 905), SHOW VIEW (block 910), SHOW TRIGGER (block 915), SHOW MACRO (block 920), and SHOW JOIN INDEX (block 925).

[0049] Finally, the portion of the database structure may be rebuilt from the database's retrieved object definitions, and optionally, the statistical view of the data, and the configuration information concerning the system (block 625). The statistical view of the data and the configuration information are provided by other portions of the system not described herein. After the structure is rebuilt, it may be necessary to populate the database with data before running the query.

[0050] In addition to being useful for troubleshooting a failed query, this method may also be helpful to a database administrator who wishes, for example, to quickly retrieve multiple table definitions to analyze a particular query.

[0051] This method can also be implemented on massively parallel processing (MPP) systems, which allow for more complex database structures. In an MPP environment, retrieving a portion of the database structure for a query may involve hundreds of object definitions. Therefore, being able to easily retrieve this structure automatically is more critical than on simpler systems with simpler database structures.

[0052] While particular preferred embodiments of the present invention have been described, it is not intended that these details should be regarded as limitations on the present invention, except to the extent that they are included in the following claims.